A Framework For Security Analysis of Key Derivation Functions

Chuah Chai Wen, Edward Dawson, Juan Manuel González Nieto, and Leonie Simpson

Queensland University of Technology,
chaiwen.chuah, e.dawson, j.gonzaleznieto, lr.simpson@qut.edu.au

Abstract

This paper presents a comprehensive formal security framework for key derivation functions (KDF). The major security goal for a KDF is to produce cryptographic keys from a private seed value where the derived cryptographic keys are indistinguishable from random binary strings. We form a framework of five security models for KDFs. This consists of four security models that we propose: Known Public Inputs Attack (KPM, KPS), Adaptive Chosen Context Information Attack (CCM) and Adaptive Chosen Public Inputs Attack(CPM); and another security model, previously defined by Krawczyk [6], which we refer to as Adaptive Chosen Context Information Attack(CCS). These security models are simulated using an indistinguisibility game. In addition we prove the relationships between these five security models and analyse KDFs using the framework (in the random oracle model).

Keywords: Key derivation function, Security framework, Indistinguishability, Cryptographic keys.

1 Introduction

Cryptographic keys are necessary for safeguarding electronic transactions, communications, and data storage. Key derivation functions (KDF) are the standard algorithm used to generate these cryptographic keys. KDFs are used to generate one or more cryptographic keys from a private seed value, such as a password, Diffie-Hellman (DH) shared secret or some non-uniformly random source material[5,7,8]. It is critical in the design of security systems that KDF proposals themselves are secure. Significant effort in designing a KDF proposal and comprehensive security analysis to evaluate the proposal is justified. The practical importance of KDFs is reflected in their adoption in industrial standard documents; for example PKCS5 [4], ISO-18033-2[9] and more recently in NIST 800-135[3]. There are KDF proposals such as [1,4,6,10] based on cryptographic hash functions.

In the current literature, formal models for the security analysis of KDFs have been introduced by Yao & Yin in [10] and Krawczyk in [6]. However, there are some limitations with each of these security models as they do not completely

cover the range of realistic capabilities of the adversary. This has motivated us to extend the existing security models into a new security framework.

In this paper we develop a comprehensive, formal security framework to form a basis for the design and analysis of KDFs. We begin with an overview of the KDF construction and define the security of KDFs in terms of an indistinguishability game. We develop a framework in which the security can be asserted in terms of the ability of adversaries of varying capabilities to win these indistinguishability games. The adversaries considered range from passive observers of information to active adversaries of varying strength.

2 Key Derivation Functions

Generally, a key derivation function KDF is defined as

$$K \leftarrow KDF(PrivS, s, ctx, n)$$

where

- $PrivS$ is a private seed. The space of all possible private seeds is denoted by $PSPACE$ and the probability distribution of $PrivS$ is assumed to be public;
- s is a salt, a public random string chosen from the salt space $SSPACE$;
- ctx is a public context string chosen chosen from a context space $CSPACE$;
- n is a positive integer that indicates the number bits of the to be produced by the KDF;
- K is the derived cryptographic key of length n bits.

Note that all inputs are publicly known, except for the secret seed $PrivS$. The salt is uniformly random and is used to create a large set of possible keys corresponding to a given private seed value. Context information is arbitrary but application specific data; for example, a session identifier or the identities of communicating parties. The basic operation of a KDF is to transform the private seed value and public inputs into an n bit pseudorandom string which can be used as a cryptographic key. The length, n, of the cryptographic key is an application specific security parameter. From now on we will represent the key derivation function as $KDF(PrivS, s, ctx)_n$.

3 General Security Framework

The general security framework is based on an indistinguishability game played between a challenger C and an adversary A in polynomial time t, where the KDF is considered secure if no A can win the game with probability significantly greater than the probability of winning by guessing randomly. To win the game A has to determine if the challenge output given in the game is the cryptographic key generated by the KDF or a truly random binary string of the same length within a polynomial number of time steps. The game runs in two major stages: the learning stage and the challenge stage. An optional stage

called the adaptive stage may be available for some powerful A, who can repeat the learning stage after receiving the challenge output. An explanation of how this game is conducted follows.

- **Learning stage**: A private seed value $PrivS$ is chosen from $PSPACE$. A can make at most q queries, either $q < |SSPACE| \times |CSPACE| < |PSPACE|$ or $q < |CSPACE| < |PSPACE|$ depending on the type of security models. For each query, a derived cryptographic key associated with a salt and context information is provided to A. A can use this information to construct a lookup table to be used in the challenge stage of the game. The capabilities of the adversary determine the level of control they have over the public inputs to KDF. A passive adversary is just an observer that obtains the cryptographic key K, but cannot query the KDF to generate a cryptographic key from their choice of public inputs. An active adversary is able to interact with the KDF to demand cryptographic keys corresponding to their choice of public inputs, with the ability to choose either salt or context information, or both.
- **Challenge stage**: A random bit $b \in_R \{0, 1\}$ is generated by C. If $b = 0$, then C computes $K' = \mathrm{KDF}(PrivS, s, ctx)$, else C outputs a random binary string K' of length of n bits. An active A may have the ability to choose either salt or context information, or both, to obtain the challenge output but this is subject to the restriction that the chosen set of public inputs were not a set of inputs from the learning stage. C sends K' to A.
- **Adaptive stage**: Give the challenge output K', a powerful active A may have the capability to learn more about K' in an adaptive stage before guessing whether K' is the cryptographic key or a binary random string. The adaptive stage consists of repeating the steps in the learning stage for up to q - q' queries, subject to the restriction that A may not ask anything directly regarding the public inputs from the challenge stage.

To complete the game, A guesses whether K' is the key or a random string. If A guesses that K' is a cryptographic key then A sends $b' = 0$, otherwise, A sends $b' = 1$. A wins the game if $b' = b$.

If the adversary is unable to distinguish between a cryptographic key derived from a private seed value using the KDF and a random string of the same length, then the KDF is secure in terms of indistinguishability. Formally, we say that the KDF is (t, q, ϵ)-secure if the probability of the adversary winning the game in time at most t with at most q queries is $\Pr[b = b'] \leq \frac{1}{2} + \epsilon$, where ϵ is negligible. If the adversary is able to distinguish the challenge output with a probability greater than $\frac{1}{2}$, then the adversary is considered to have an 'advantage' in distinguishing the cryptographic keys which are produced by the KDF and KDF is considered insecure.

4 Defining the Security Models

The major security goal for a KDF is that the cryptographic keys generated by the KDF are indistinguishable from truly random binary strings of the same

length. That is, this KDF's security goal is formalized as an adversary's inability to gain any information about cryptographic keys derived from a private seed value, even though public inputs are provided to the adversary. We consider this security goal in situations where the capability of the adversary differs and use this to establish five security models: KPM, KPS, CCM, CCS and CPM. Two models, KPM and KPS, are weak security models as A is only an observer. The other models, CCM, CCS and CPM, are stronger security models as the adversary is active. The difference between these three security models lies in the capability of A in choosing the public inputs. For CCM and CCS, A can only choose ctx while A can choose both s and ctx in CPM. Table 1 briefly summarizes the capability of the adversary in the five security models. The symbol '$\surd$' indicates that the adversary is able to query the KDF to generate the cryptographic keys from their choice of public inputs. The symbol 'X' indicates that the adversary is not able to choose the public inputs although these are known by the adversaries. The symbol '-' indicates that the adversaries are not able to learn more about the challenge output at the adaptive stage. Each of these security models is discussed in greater detail in the following sections.

Security Models	KPM	KPS	CCM	CCS	CPM
Type of Adversary	Passive	Passive	Active	Active	Active
Type of Salt	Multiple	Fixed	Multiple	Fixed	Multiple
Number of Queries, $q <$	$\|SSPACE\| \times \|CSPACE\|$	$\|CSPACE\|$	$\|SSPACE\| \times \|CSPACE\|$	$\|CSPACE\|$	$\|SSPACE\| \times \|CSPACE\|$
Capability A in choosing:					
Learning Stage					
Salt	X	X	X	X	$\surd$
Context information	X	X	$\surd$	$\surd$	$\surd$
Challenge Stage					
Salt	X	X	X	X	$\surd$
Context information	X	X	$\surd$	$\surd$	$\surd$
Adaptive Stage					
Salt	-	-	X	X	$\surd$
Context information	-	-	$\surd$	$\surd$	$\surd$

Table 1. Summary of the capabilities of the adversary in five security models.

4.1 Known Public Inputs Attack - KPM-secure

For the KPM security model, the adversary can observe the salt and context information, and the resulting derived key. At the learning stage, each cryptographic key is generated from a fixed private seed value together with a different salt, and with the same or different context information. These cryptographic keys are provided to adversaries. At the challenge stage, A is presented with a binary string. If the KDF is secure, A should not be able to distinguish whether this string is a cryptographic key or a random string of the same length.

Definition 1 {**KPM-secure**} *The KDF is (t, q, ϵ) KPM-secure if for all adversaries A running in polynomial time t and making at most $q < |SSPACE| \times |CSPACE|$ queries to the KDF with known multiple salt and known context information win the following indistinguishability game with probability not larger than ($\frac{1}{2} + \epsilon$).*

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. For $i = 1, \ldots, q' \leq q$,	 (2.1) C chooses $s_i \xleftarrow{R} SSPACE$ and $ctx_i \leftarrow CSPACE$. (2.2) C computes $K_i = F(PrivS, s_i, ctx_i)_n$. (2.3) A is provided with the triple (K_i, s_i, ctx_i).
Challenge stage	1. C chooses $s \xleftarrow{R} SSPACE$ and $ctx \leftarrow CSPACE$ 2. C chooses $b \xleftarrow{R} \{0, 1\}$. 3. C sends K', s and ctx to A. 4. A outputs $b' = 0$, if $\mathcal{A}$ believes that K' is cryptographic key, else outputs $b' = 1$.	 (2.1) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$, (2.2) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
A wins the game if $b' = b$.		

4.2 Known Public Inputs Attack - KPS-secure

For the KPS, the adversary is an observer. In this indistinguishability game, each cryptographic key is generated from a fixed private seed value together with a fixed salt, and with the different context information. The major difference between KPM-secure and KPS-secure is that for KPM-secure, multiple salts are used to generate the cryptographic keys while for KPS-secure, a fixed salt is used for generating one or more cryptographic keys.

Definition 2 {**KPS-secure**} *The KDF is (t, q, ϵ) KPS-secure if for all adversaries A running in polynomial time t and making at most $q < |CSPACE|$ queries to the KDF with known fixed salt and known context information win the following indistinguishability game with probability not larger than $(\frac{1}{2} + \epsilon)$.*

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. C chooses $s \xleftarrow{R} SSPACE$. 3. A is provided with the value s. 4. For $i = 1, \ldots, q' \leq q$,	 (4.1) C chooses $ctx_i \leftarrow CSPACE$. (4.2) C computes $K_i = F(PrivS, s, ctx_i)_n$. (4.3) A is provided with the pair (K_i, ctx_i).
Challenge stage	1. C chooses $ctx \leftarrow CSPACE$ (subject to restriction $ctx \notin ctx_i, \ldots, ctx'_q$). 2. C chooses $b \xleftarrow{R} \{0, 1\}$. 3. C sends K' and ctx to A. 4. A outputs $b' = 0$, if $\mathcal{A}$ believes that K' is cryptographic key, else outputs $b' = 1$.	 (2.1) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$, (2.2) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
A wins the game if $b' = b$.		

4.3 Adaptive Chosen Context Information Attack (CCM)

For the CCM model, the adversaries are active, and are capable of choosing the context information in the indistinguishability game. For CCM, the adversaries are allowed to query multiple context information used with the same private seed value and with different randomly generated salt to form the cryptographic keys.

Definition 3 {**CCM-secure**} *The KDF is (t, q, ϵ) CCM-secure if for all adversaries A running in polynomial time t and making at most $q < |SSPACE| \times$*

$|CSPACE|$ queries to the KDF with known multiple salt and chosen context information win the following indistinguishability game with probability not larger than $(\frac{1}{2} + \epsilon)$.

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$.	
	2. For $i = 1, \ldots, q' \leq q$,	(2.1) C chooses $s_i \xleftarrow{R} SSPACE$. (2.2) A is provided s_i. (2.3) A chooses $ctx_i \leftarrow CSPACE$. (2.4)C computes $K_i = F(PrivS, s_i, ctx_i)_n$. (2.5)$A$ is provided the derived cryptographic key, K_i.
Challenge stage	1. C chooses $s \xleftarrow{R} SSPACE$. 2. A is provided s. 3. A chooses $ctx \leftarrow CSPACE$. 4. C chooses $b \xleftarrow{R} \{0, 1\}$. 5. C sends K' to A.	(4.1) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$, (4.2) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
Adaptive stage	1. Step 2 in **Learning stage** is repeated for up to $q - q'$ queries (subject to restriction $\{s_i, ctx_i\} \neq \{s, ctx\}$). 2. A outputs $b' = 0$, if $\mathcal{A}$ believes that K' is cryptographic key, else outputs $b' = 1$.	
A wins the game if $b' = b$.		

4.4 Adaptive Chosen Context Information Attack (Krawczyk)

The formal security model for KDFs proposed by Krawczyk [6] is included in our framework. We refer to this model as CCS-secure. For this security model, the adversaries are capable of influencing the inputs in the indistinguishability game, and are allowed to query multiple context information under the same private seed value with the same randomly generated salt.

Definition 4 {**CCS-secure**} *The KDF is (t, q, ϵ) CCS-secure if for all adversaries A running in polynomial time t and making at most $q < |CSPACE|$ queries to the KDF with known fixed salt and chosen context information win the following indistinguishability game with probability not larger than $(\frac{1}{2} + \epsilon)$.*

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. C chooses $s \xleftarrow{R} SSPACE$. 3. A is provided with the value s. 4. For $i = 1, \ldots, q' \leq q$,	(4.1) A chooses $ctx_i \leftarrow CSPACE$. (4.2) C computes $K_i = F(PrivS, s, ctx_i)_n$. (4.3)$A$ is provided the derived cryptographic key, K_i.
Challenge stage	1. A chooses $ctx \leftarrow CSPACE$ (subject to restriction $ctx \notin ctx_i, \ldots, ctx'_q$). 2. C chooses $b \xleftarrow{R} \{0, 1\}$. 5. C sends K' to A.	(2.1) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$, (2.2) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
Adaptive stage	1. Step 4 in **Learning stage** is repeated for up to $q - q'$ queries (subject to restriction $ctx_i \neq ctx$). 2. A outputs $b' = 0$, if $\mathcal{A}$ believes that K' is cryptographic key, else outputs $b' = 1$.	
A wins the game if $b' = b$.		

4.5 Adaptive Chosen Public Inputs Attack (CPM)

The Krawczyk security model restricts the capability of the strong active adversary. The adversary is only able to change the context information. In some

situations, an active adversary may exist that can influence all the possible inputs for KDFs: the salt and the context information, as shown in [2]. This situation motivated the creation of a security model called CPM-secure. For a KDF to be CPM-secure, an adversary A who is allowed to choose both public inputs, salt and context information. For instance, the adversary may choose a null or non-random salt value. The adversary's chosen salt value and different chosen context information is used to generate the cryptographic keys. The adversaries are able to choose whether to respond to the challenger immediately or to progress to the adaptive stage. Again the adversaries are allowed to make no more than q queries.

Definition 5 {**CPM-secure**} *The KDF is (t, q, ϵ) CPM-secure if for all adversaries A running in polynomial time t and making at most $q < |SSPACE| \times |CSPACE|$ queries to the KDF with chosen salt and chosen context information win the following indistinguishability game with probability not larger than ($\frac{1}{2} + \epsilon$).*

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. For $i = 1, \ldots, q' \leq q$,	 (2.1) A chooses $s_i \leftarrow SSPACE$ and $ctx_i \leftarrow CSPACE$. (2.2) C computes $K_i = F(PrivS, s_i, ctx_i)_n$. (2.3) A is provided the derived cryptographic key, K_i.
Challenge stage	1. A chooses $s \leftarrow SSPACE$ and $ctx \leftarrow CSPACE$. (subject to restriction $\{s, ctx\} \notin \{s_i, ctx_i\}, \ldots, \{s'_q, ctx'_q\}$). 2. C chooses $b \overset{R}{\leftarrow} \{0, 1\}$. 3. C sends K' to A.	 (2.1) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$, (2.2) else C outputs $K' \overset{R}{\leftarrow} \{0,1\}^n$.
Adaptive stage	1. Step 2 in **Learning stage** is repeated for up to $q - q'$ queries (subject to restriction $\{s_i, ctx_i\} \neq \{s, ctx\}$). 2. A outputs $b' = 0$, if $\mathcal{A}$ believes that K' is cryptographic key, else outputs $b' = 1$.	
A wins the game if $b' = b$.		

5 Relating These Five Security Models

The models described above provide assurance for varying levels of security. A KDF which is considered secure under one model may not be under another. For example, a KDF may be KPM-seucre but not CPM-secure. In this section, we establish more precisely the relations between these five security models. Figure 1 gives a summary of these relations.

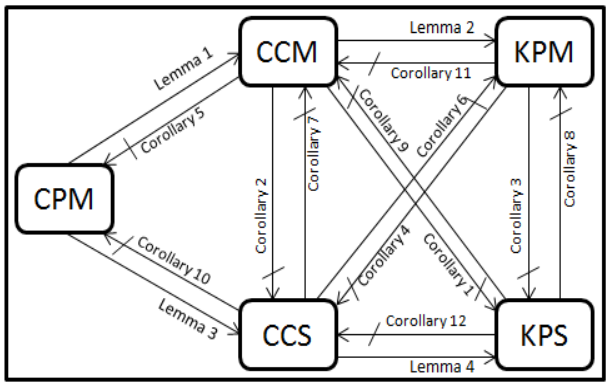

Fig. 1. The relationship between the proposed five security models.

5.1 Implications between Security Models

We start by studying the implication relationships between the different security notions. These are shown as lemmas 1-4 in Figure 1.

Lemma 1 *CPM* $\Rightarrow$ *CCM*

Proof: Assume a KDF is CPM-secure but not CCM-secure. Since the KDF is not CCM-secure, then there exists an adversary A who can win the CCM game with probability greater than $\frac{1}{2} + \epsilon$. Now, we assume an adversary B who plays the CPM game with C. B will make use of the capability of A, so that A is playing the CCM game with B while B is playing the CPM game with C.

The game is conducted as below:

- **Learning stage**
  1. C chooses $PrivS \leftarrow PSPACE$.
  2. For $i = 1, \ldots, q' \leq q$,
     (a) B chooses $s_i \leftarrow SSPACE$ and sends it over to A.
     (b) A chooses $ctx_i \leftarrow CSPACE$ and sends it over to B.
     (c) B forwards s_i and ctx_i to C. C computes $K_i = F(PrivS, s_i, ctx_i)_n$.
     (d) B is provided K_i. B forwards K_i to A.
- **Challenge stage**
  1. B chooses $s \leftarrow SSPACE$ and forwards s to A.
  2. A chooses $ctx \leftarrow CSPACE$.A sends ctx to B.
  3. B forwards s and ctx to C.
  4. C chooses $b \xleftarrow{R} \{0, 1\}$.
     (a) If $b = 0$, C outputs $K' = F(PrivS, s, ctx)_n$,
     (b) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
  5. C sends K' to B and B forwards K' to A.
- **Adaptive stage**
  1. Step 2 in **Learning stage** is repeated for up to $q - q'$ queries (subject to restriction $\{s_i, ctx_i\} \neq \{s, ctx\}$).

2. A outputs $b' = 0$, if A believes that K' is cryptographic key, else outputs $b' = 1$. A sends b' to B and B simply forwards b' to C.
3. B wins the game if $b'_A = b_C$.

The probability that B wins the CPM game is equal to the probability that A wins the CCM game. Our assumption is that the KDF is not CCM-secure. That is the probability that A wins the CCM game is greater than $\frac{1}{2} + \epsilon$. Therefore, B wins the CPM game with probability greater than $\frac{1}{2} + \epsilon$. This implies that the KDF is not CPM-secure. This is a contradiction. Hence, $CPM \Rightarrow CCM$. □

Lemma 2 $CCM \Rightarrow KPM$

Lemma 3 $CPM \Rightarrow CCS$

Lemma 4 $CCS \Rightarrow KPS$

The proofs of these lemmas are similar to the proof of Lemma 1.

5.2 Non-implications between Security Models

To prove the non-implications between the security models (corollaries 1-12 in Figure 1), we analyse five KDFs, $KDF1$-$KDF5$, all based on an underlying hash function F (Table 2). Of these KDFs, three are proposals found in the literature. The other two are (contrived) KDF designs which are useful to demonstrate the separation between some of the security models. Perhaps the most interesting observation from our results in this section is that security when the salt value is fixed does not imply security when different salt values are used.

In what follows, all the proofs are given in the random oracle model (ROM). While proofs in the standard model would be clearly preferable, we believe that using the ROM is appropriate for our purposes. Firstly, as observed by others [6,10], many hash-based KDFs proposed in the literature and used in standards seem impossible to be proven secure based on standard properties of the underlying hash functions. Yet one would like to show that these "practical" hash-based KDFs have some level of security that justifies their use. For example, $KDF1$ in Table 2, which is standardised in PKCS#5 [4], does not seem provable without considering idealised properties of the underlying hash function. An extensive discussion on the applicability of the ROM in the analysis of KDFs is given by Krawczyk [6]. In addition, use of the ROM in this work is sufficient for our purpose of studying the relationships between the different security notions in our framework.

Theorem	KDF proposals	KPM	KPS	CCM	CCS	CPM
1	$KDF1(PrivS, s, ctx)_n = F^{ctx}(PrivS, s)$, assuming ctx is an integer value [4]	√	X	√	X	X
2	$KDF2(PrivS, s, ctx)_n = F(PrivS\|\|ctx)\|\|F(PrivS\|\|s\|\|ctx)$	X	√	X	√	X
3	$KDF3(PrivS, s, ctx)_n = F(PrivS\|\|s\|\|ctx_1)\|\| F(PrivS\|\|s\|\|ctx_2)$	√	√	X	X	X
4	$KDF4(PrivS, s)_n = F(PrivS, s_1, PrivS)\|\| F(PrivS, s_2, PrivS)\|\| \ldots \|\| F(PrivS, s_l, PrivS)$, where $s = s_1 \,\|\| \, s_2 \,\|\| \ldots \|\| \, s_l$ [1]	√	√	√	√	X
5	$KDF5(PrivS, s, ctx)_n = F(F(PrivS, s), ctx)$ [6]	√	√	√	√	√

Table 2. Summary of the security analysis of KDF proposals based on the proposed formal security framework for KDF.

KDF1 Here we analyse the security of *KDF1*, which corresponds to *PBKDF1*, a password-based KDF standardised in PKCS#5 [4]. *KDF1* is defined as

$$KDF1(PrivS, s, ctx)_n = F^{ctx}(PrivS, s),$$

where the context ctx is an integer value which indicates the number of iterations of the hash function F. We show that it achieves CCM security. In addition, we use the analysis of *PBKDF1* to prove the non-implications of corollaries 1-5 in Figure 1.

Theorem 1 *KDF*1 *is secure with the respect to KPM, CCM and is not secure in KPS, CCS and CPM.*

Proof: Firstly, we show that *KDF1* is CCM secure. The proof is in the RO model, where in order to obtain the value $F(x)$, the adversary needs to query the random oracle with input x. The random oracle queries are simulated by the challenger as follows. On input a string x, if x has not been queried before, output $F(x) \in_R \{0,1\}^n$, where n is the output length of the hash function. If x has been queried before, output the same value $F(x)$ as before. Let q_F and q_k be the number of queries made by the adversary to the random oracle and the KDF oracle, respectively.

During the learning stage, C chooses $PrivS \in_R PSPACE$, $s_i \in_R SSPACE$ and uses F to compute $K_i = F^{ctx_i}(PrivS, s_i)$, where A chooses. At the challenge stage, a challenge key is computed as $K' = F^{ctx}(PrivS, s)$, where C chooses s and A chooses ctx. A receives K' and continues to learn cryptographic keys by making up to $q - q'$ KDF queries during the adaptive stage.

Since, F is modeled as a RO, A can only distinguish whether K' is the cryptographic key generated from *KDF1* or is a random key with probability different from $\frac{1}{2}$, if one of the following happens:

a) $s = s_i$ for some $i = 1 \dots q_k$. This implies that s was chosen during the learning stage. When this occur, the adversary can easily win the game as follows. Without loss of generality, assume $s = s_i$ and $ctx \leq ctx_i$. Then, $F^{ctx_i - ctx}(PrivS, s) = K_i$, which can be checked by the adversary.
The probability that this case occurs is $Pr[s = s_i] = \frac{q_k}{|SSPACE|}$.

b) The adversary queries $F(PrivS'||s')$ to the random oracle, such that $PrivS'||s' = PrivS||s$. This amounts to A guessing $PrivS$, which can happen with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A wins this indistinguishability game is:

$$\begin{aligned}
\Pr[A \text{ wins}] = {} & \Pr[A \text{ wins}|s = s_i]\Pr[s = s_i] + \\
& \Pr[A \text{ wins}|PrivS' = PrivS]\Pr[PrivS' = PrivS] + \\
& \Pr[A \text{ wins}|s \neq s_i \wedge PrivS' \neq PrivS]\Pr[s \neq s_i \wedge PrivS' \neq PrivS] \\
\leq {} & 1\left(\frac{q_k}{|SSPACE|}\right) + 1\left(\frac{q_F}{|PSPACE|}\right) + \frac{1}{2}\left(1 - \frac{q_k}{|SSPACE|} - \frac{q_F}{|PSPACE|}\right) \\
\leq {} & \frac{1}{2} + \frac{q_k}{2|PSPACE|} + \frac{q_F}{2|PSPACE|},
\end{aligned}$$

where $\epsilon = \frac{q_k}{2|SSPACE|} + \frac{q_F}{2|PSPACE|}$ is negligible.

A only has negligible 'advantage' over random guessing the challenge output. Therefore, *KDF1* is CCM-secure. *KDF1* is KPM-secure by Lemma 2.

Secondly, we show *KDF1* is not secure in CCS. Recall from Definition 4, that in CCS, the salt s is fixed for the entire indistinguishability game and A is allowed to choose ctx.An attack to the CCS security of *KDF1* is as follows. In the learning stage, A queries ctx_1 to get K_1, such as $K_1 = F^{ctx_1}(PrivS,\ s)$. During the challenge stage, A asks for the challenge output corresponding to context $ctx_1 - 1$ to the KDF. A bit b is choosen randomly to output challenge output, $b = 0$, if cryptographic key, $K' = F^{ctx^{*-1}}(PrivS,\ s)$ or $b = 1$, $K' =$ random string. Once A receives the value of K', A checks $K_1 \stackrel{?}{=} F(K')$. If so, A outputs $b' = 0$, otherwise $b' = 1$. A wins the game as $b' = b$ except with negligible probability $\frac{1}{2^n}$ (corresponding to the case where $b = 1$, but still $K_1 = F(K')$).

Next we show that *KDF1* is not KPS-secure. The attack is similar to the strategy followed by the adversary in the CCM game above in the case where $s = s_i$ for some $i = 1, \ldots, q_k$. In the KPS game the salt s is fixed and the context is chosen by the challenger. Without loss of generality assume that $ctx_i \leq ctx$ for some $i = 1, \ldots, q_k$. The adversary checks if $K' = F^{ctx - ctx_1}(K_i)$. If so, it outputs $b' = 0$, else outputs $b' = 1$. Again, we see that the adversary wins with all but negligible probability.

Finally, it follows from Lemma 3 that since *KDF1* is not CCS-secure, then it is not CPM secure. □

The proof of Corollary 1 - 5 are an immediate result of Theorem 1:
Corollary 1 $CCM \nrightarrow KPS$ If a KDF is CCM-secure, it may not be KPS-secure.
Corollary 2 $CCM \nrightarrow CCS$ If a KDF is CCM-secure, it may not be CCS-secure.
Corollary 3 $KPM \nrightarrow KPS$ If a KDF is KPM-secure, it may not be KPS-secure.
Corollary 4 $KPM \nrightarrow CCS$ If a KDF is KPM-secure, it may not be CCS-secure.
Corollary 5 $CCM \nrightarrow CPM$ If a KDF is CCM-secure, it may not be CPM-secure.

KDF2 Here we analyse the security of *KDF2* which is defined as,

$$KDF2(PrivS,\ s,\ ctx)_n = F(PrivS \| ctx) \| F(PrivS \| s \| ctx)$$

We show that it achieves CCS security. Furthermore, we use the analysis of *KDF2* to prove the non-implications of corollaries 6-10 in Figure 1.

Theorem 2 *$KDF2$ is secure with respect to KPS, CCS and is not secure in KPM, CCM and CPM.*

Proof: Firstly, we prove that $KDF2$ is CCS-secure. Again, the proof is in the RO model as in Theorem 1. In the learning stage, C chooses $PrivS \in_R PSPACE$, $s \in_R SSPACE$ and C uses F to compute $K_i = KDF2(PrivS, s, ctx_i)_n$, where A chooses ctx_i. At the challenge stage, challenge key is computed as $K' = KDF2(PrivS, s, ctx)_n$, where s is same as at the learning stage and ctx is chosen

by A. C sends K' to A. Once A receives K', A continues learn the cryptographic keys which are derived from $KDF2$ up to $q - q'$ queries.

Since, F is modeled as a RO, hence, A can only distinguish if K' is the key generated by $KDF2$ or a random string of the same length, only if:

a) $ctx = ctx_i$ for some $i = 1 \dots q_k$. In this case, $F(PrivS \| ctx) \| F(PrivS \| s \| ctx) = F(PrivS \| ctx_i) \| F(PrivS \| s \| ctx_i)$, it means $K' = K_i$, where K_i is one of the key at the learning stage. Hence, A can distinguish K' is the key generated by $KDF2$ by checking that K' is one of the key which had been generated at the learning stage. However, recall Definition 4, during the challenge stage, A is not allow to choose $ctx = ctx_i \forall i$ which had been chosen at the learning stage. It means, $ctx \neq ctx_i \forall i$. Consequently, the probability is $\Pr[ctx = ctx_i] = 0$.
b) Query $PrivS$ such as $PrivS' = PrivS$ and find $F(PrivS' \| ctx) = F(PrivS \| ctx)$ or $F(PrivS' \| s \| ctx) = F(PrivS \| s \| ctx)$. This amounts to A guessing $PrivS$, which can happen with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A winning this indistinguishability game is:

$$\begin{aligned}\Pr[A \text{ wins}] &= \Pr[A \text{ wins} | PrivS' = PrivS]\Pr[PrivS' = PrivS] + \\ &\quad \Pr[A \text{ wins} | PrivS' \neq PrivS]\Pr[PrivS' \neq PrivS] \\ &\leq 1\left(\frac{q_F}{|PSPACE|}\right) + \frac{1}{2}\left(1 - \frac{q_F}{|PSPACE|}\right) \\ &\leq \frac{1}{2} + \frac{q_F}{2|PSPACE|}, \text{ where } \epsilon = \frac{q_F}{2|PSPACE|} \text{ is negligible.}\end{aligned}$$

A only has negligible 'advantage' over random guessing the challenge output. Therefore, this KDF is CCS-secure. $KDF2$ is KPS-secure by Lemma 4.

Secondly, we show $KDF2$ is not KPM-secure. Recall from the Definition 1, the KDF will be KPM-secure if after making q queries where $q < |SSPACE| \times |CSPACE|$ at the learning stage and at the challenge stage, A is not able to distinguish between a key produced by the KDF and a truly random binary string with the same length with probability not greater than $\frac{1}{2} + \epsilon$, where probability ϵ is negligible.

During the learning stage, C chooses $p \in_R PSPACE$, $s \in_R SSPACE$, $c_i \in CSPACE$ and uses F to compute $K_i = KDF2(p, s_i, c_i, n)$. A receives K_i, s_i and c_i. In the challenge stage, challenge key K' is computed by $K' = KDF2(p, s, c, n)$, where $s \in_R SSPACE$ and $c \in CSPACE$.

Since F is modeled as a ROM, A can distinguish if K' is the key generated by $KDF2$ or a random string of the same length, only if C chooses $c = c_i$ for some $i = 1 \dots q_k$. This implies that c has been chosen at the learning stage, where the first half of K' is similar with the first half of K_i for some $i = 1 \dots q_k$ at the learning stage. A can distinguish the challenge output is cryptographic key by observing the first component part of K'. Since set space of s is greater than set space of c, therefore, c will be reused, the probability is, $Pr[c = c_i] = 1$.

Therefore, $KDF2$ is not KPM-secure. Hence, $KDF2$ is not CCM-secure by Lemma 2 and is not CPM-secure by Lemma 1. □

The proof of Corollary 6 - 10 are an immediate result of Theorem 2:

Corollary 6 *CCS* $\nrightarrow$ *KPM* If a KDF is CCS-secure, it may not be KPM-secure.

Corollary 7 *CCS* $\nrightarrow$ *CCM* If a KDF is CCS-secure, it may not be CCM-secure.
Corollary 8 *KPS* $\nrightarrow$ *KPM* If a KDF is KPS-secure, it may not be KPM-secure.
Corollary 9 *KPS* $\nrightarrow$ *CCM* If a KDF is KPS-secure, it may not be CCM-secure.
Corollary 10 *CCS* $\nrightarrow$ *CPM* If a KDF is CCS-secure, it may not be CPM-secure.

KDF3 Now we analyse the security of *KDF3*, which we defined it as,

$$KDF3(PrivS,\ s,\ ctx)_n = F(PrivS\|s\|ctx_1)\|\ F(PrivS\|s\|ctx_2), \text{ where } ctx = ctx_1\|ctx_2.$$

We show that it only achieves KPS and KPM security. In addition, we use this analysis to prove the non-implications of corollaries 11 and 12 in Figure 1.

Theorem 3 *KDF*3 *is secure with respect to KPM, KPS and is not CCM, CCS and CPM.*

Proof: Firstly, we show that *KDF*3 is KPM-secure. During the learning stage, C chooses $PrivS \in_R PSPACE$, $s_i \in_R SSPACE$, $ctx_i \in CSPACE$ and C uses F to compute $K_i = KDF3(PrivS, s_i, ctx_i)_n$. A is provided K_i, s_i and ctx_i. In the challenge stage, challenge key is computed as $K' = KDF3(PrivS, s, ctx)_n$, where $s \in_R SSPACE$ and ctx is chosen by C.

Since F is modeled as a RO, A can only distinguish if K' is the key generated by *KDF*3 or a random string of the same length, only if:

a) C chooses $ctx \in CSPACE$, $ctx = ctx_1\|ctx_2$ and $ctx_1 = ctx_2$. In this case, A can check the challenge key K', the first half is equal with the second half of K', such as $F(PrivS\|s\|ctx_1) = F(PrivS\|s\|ctx_2)$. However, with a high probability the chosen ctx by C is most likely different such that $ctx_1 \neq ctx_2$. Hence, the probability is, $\Pr[ctx_1 = ctx_2] \leq \frac{1}{|CSPACE|}$.
b) $s = s_i$ and $ctx = ctx_i$ for some $i = 1 \ldots q_k$. In this case, A will distinguish K' is one of the cryptographic key at the learning stage, such as $K' = K_i$ for some $i = 1 \ldots q_k$, where $F(PrivS\|s\|ctx_1)\|F(PrivS\|s\|ctx_2) = F(PrivS\|s_i\|ctx_1^i)\|F(PrivS\|s_i\|ctx_2^i)$. The probability is, $\Pr[s = s_i \wedge ctx = ctx_i] \leq \frac{q_k}{|SSPACE|\times|CSPACE|}$.
c) Query $PrivS$, $PrivS' = PrivS$ and find $F(PrivS'\|s\|ctx_1) = F(PrivS\|s\|ctx_1)$ or $F(PrivS'\|s\|ctx_2) = F(PrivS\|s\|ctx_2)$. This implies that A guessing $PrivS$ which can happen with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A winning the game is:

$$\begin{aligned}
\Pr[A \text{ wins}] = {} & \Pr[A \text{ wins}|ctx_1 = ctx_2]\Pr[ctx_1 = ctx_2] + \\
& \Pr[A \text{ wins}|s = s_i \wedge ctx = ctx_i]\Pr[s = s_i \wedge ctx = ctx_i] + \\
& \Pr[A \text{ wins}|PrivS' = PrivS]\Pr[PrivS' = PrivS] + \\
& \Pr[A \text{ wins}|ctx_1 \neq ctx_2 \wedge s \neq s_i \& ctx \neq ctx_i \wedge PrivS' \neq PrivS]\times \\
& \Pr[ctx_1 \neq ctx_2 \wedge s \neq s_i \& ctx \neq ctx_i \wedge PrivS' \neq PrivS] \\
\leq {} & 1\left(\tfrac{1}{|CSPACE|}\right) + 1\left(\tfrac{q_k}{|SSPACE||CSPACE|}\right) + 1\left(\tfrac{q_F}{|PSPACE|}\right) + \\
& \tfrac{1}{2}\left(1 - \tfrac{1}{|CSPACE|} - \tfrac{q_k}{|SSPACE||CSPACE|} - \tfrac{q_F}{|PSPACE|}\right)
\end{aligned}$$

$\leq \frac{1}{2} + \frac{1}{2|CSPACE|} + \frac{q_k}{2|SSPACE||CSPACE|} + \frac{q_F}{2|PSPACE|}$,
where $\epsilon = \frac{1}{2|CSPACE|} + \frac{q_k}{2|SSPACE||CSPACE|} + \frac{q_F}{2|PSPACE|}$ is negligible.

A only has negligible 'advantage' over random guessing the challenge output. Therefore, $KDF3$ is KPM-secure.

Secondly, we show that $KDF3$ is KPS-secure. During the learning stage, C chooses $PrivS \in_R PSPACE$. C choose a fixed $s \in_R SSPACE$ which is used for entire game and chooses different $ctx_i \in CSPACE$. Then, C uses F to compute $K_i = KDF3(PrivS, s, ctx_i)_n$. A is provided K_i, s and ctx_i. In the challenge stage, challenge key is computed as $K' = KDF3(PrivS, s, ctx)_n$, where ctx is chosen by C.

Since F is modeled as a RO, A can only distinguish if K' is the key generated by $KDF3$ or a random string of the same length, only if:

a) C chooses $ctx \in CSPACE$, $ctx = ctx_1 \| ctx_2$ and $ctx_1 = ctx_2$. Hence, A can distinguish the challenge key K' such as the first half of the K' is equal with the second half of the K', where $F(PrivS \| s \| ctx_1) = F(PrivS \| s \| ctx_2)$. However, the chosen ctx by C is most likely different such that $ctx_1 \neq ctx_2$. Therefore, the probability if $\Pr[ctx_1 = ctx_2] \leq \frac{1}{|CSPACE|}$.
b) Query $PrivS$, $PrivS' = PrivS$ and find $F(PrivS' \| s \| ctx_1) = F(PrivS \| s \| ctx_1)$ or $F(PrivS' \| s \| ctx_2) = F(PrivS \| s \| ctx_2)$. This implies that A guessing $PrivS$ which can happen with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A winning the game is:

$$\begin{aligned}
\Pr[A \text{ wins}] = {} & \Pr[A \text{ wins} | ctx_1 = ctx_2] \Pr[ctx_1 = ctx_2] + \\
& \Pr[A \text{ wins} | PrivS' = PrivS] \Pr[PrivS' = PrivS] + \\
& \Pr[A \text{ wins} | ctx_1 \neq ctx_2 \wedge PrivS' \neq PrivS] \Pr[ctx_1 \neq ctx_2 \wedge PrivS' \neq PrivS] \\
\leq {} & 1 \left(\frac{1}{|CSPACE|} \right) + 1 \left(\frac{q_F}{|PSPACE|} \right) + \frac{1}{2} \left(1 - \frac{1}{|CSPACE|} - \frac{q_F}{|PSPACE|} \right) \\
\leq {} & \frac{1}{2} + \frac{1}{2|CSPACE|} + \frac{q_F}{2|PSPACE|},
\end{aligned}$$

where $\epsilon = \frac{1}{2|CSPACE|} + \frac{q_F}{2|PSPACE|}$ is negligible.

A only has negligible 'advantage' over random guessing the challenge output. Hence, $KDF3$ is KPS-secure.

Thirdly, we show $KDF3$ is not CCM-secure. In the learning stage, C chooses $PrivS \in_R PSPACE$ and $s_i \in_R SSPACE$, then C uses F to compute $K_i = KDF3(PrivS, s_i, ctx_i)_n$, where ctx_i is chosen by A. A receives K_i and s_i.

In the challenge stage, the challenge key K' is computed by C such as $K' = KDF3(PrivS, s, ctx)_n$, where $s \in_R SSPACE$ by C and ctx is chosen by A.

Once C sends K' to A, A continues learn the cryptographic keys $KDF3$ up to $q - q'$ queries at the adaptive stage.

F is modeled as a RO, A can only distinguish if K' is the key generated by $KDF3$ or a random string of the same length, only if, A chooses $ctx \in CSPACE$, $ctx = ctx_1 \| ctx_2$ and $ctx_1 = ctx_2$. In this case, $F(PrivS \| s \| ctx_1) = F(PrivS \| s \| ctx_2)$, A can distinguish the challenge key K' where the first half of K' is equal with the second half of K'. Since A chooses ctx, hence, the probability

is, $\Pr[ctx_1 = ctx_2] = 1$. Thus, $KDF3$ is not CCM-secure and $KDF3$ is not CPM-secure by Lemma 1.

Next, we show $KDF3$ is not CCS-secure. During the learning stage, C chooses $PrivS \in_R PSPACE$ and $s \in_R SSPACE$, which are fixed for entire game, then C uses F to compute $K_i = KDF3(PrivS, s, ctx_i)_n$, where ctx_i is chosen by A. A receives K_i and s. In the challenge stage, challenge key is computed as $K' = KDF3(PrivS, s, ctx)_n$, where ctx is chosen by A. C sends K' to A, A continues learn the cryptographic keys $KDF3$ up to $q - q'$ queries at the adaptive stage.

Since, F is modeled as a RO, A can only distinguish if K' is the key generated by $KDF3$ or a random string of the same length, only if, A chooses $ctx \in CSPACE$, $ctx = ctx_1 \| ctx_2$ and $ctx_1 = ctx_2$, A can distinguish the challenge key K' where the first half of K' is equal with the second half of K', such as $F(PrivS \| s \| ctx_1) \| F(PrivS \| s \| ctx_2)$. ctx is chosen by A, hence, the probability is, $\Pr[ctx_1 = ctx_2] = 1$. Therefore, $KDF3$ is not CCS-secure and $KDF3$ is not CPM-secure by Lemma 3. □

The proof of Corollary 11 - 12 are an immediate result of Theorem 3.
Corollary 11 *KPM $\nrightarrow$ CCM* If a KDF is KPM-secure, it may not be CCM-secure.
Corollary 12 *KPS $\nrightarrow$ CCS* If a KDF is KPS-secure, it may not be CCS-secure.

KDF4 Here we analyse the security of *KDF4*, which proposed by Adam *et. al* in 2004 [1]. *KDF4* is defined as,

$$KDF4(PrivS, s)_n = F(PrivS, s_1, PrivS) \| F(PrivS, s_2, PrivS) \| \ldots \| F(PrivS, s_l, PrivS), \text{ where } s = s_1 \| s_2 \| \ldots \| s_l.$$

We show that it achieves CCM and CCS security.

Theorem 4 *If $q = 0$ and $l > 1$, then KDF4 is secure in CCM, CCS, KPM and KPS but is not secure in CPM.*

Proof: Firstly, we prove that $KDF4$ is secure in KPM, KPS, CCM and CCS when $q = 0$ and $l > 1$ as follows. If $q = 0$, it means, A can play neither at the learning stage nor at the adaptive stage. During the challenge stage, the challenge key is computed as $K' = KDF4(PrivS, s)_n$, where $PrivS$ and s are chosen by C, such as $PrivS \in_R PSPACE$ and $s \in_R SSPACE$. K' is provided to A.

Since F is modeled as a RO, A can only distinguish if K' is the key generated by $KDF4$ or a random string of the same length, only if:

a) If the chosen s where $s = s_1 \| s_2 \| \ldots \| s_l$ and $s_1 = s_2 = \ldots = s_l$. Then, A can distinguish K' it the cryptographic key which is derived from $KDF4$ by checking is there has l repetition component parts, such as $F(PrivS, s_1, PrivS) = F(PrivS, s_2, PrivS) = \ldots = F(PrivS, s_l, PrivS)$. However, s is chosen randomly by C. Hence, the probability is $\Pr[s = s_1 \| s_2 \| \ldots \| s_l] \leq \frac{1}{|SSPACE|}$.

b) Query $PrivS$, $PrivS' = PrivS$ and find $KDF4(PrivS', s)_n = KDF4(PrivS, s)_n$. This amounts to A guessing $PrivS$, with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A winning this indistinguishability game is:

$$\begin{aligned}
\Pr[A \text{ wins}] = {} & \Pr[A \text{ wins}|s = s_1\|s_2\|\ldots\|s_l]\Pr[s = s_1\|s_2\|\ldots\|s_l] + \\
& \Pr[A \text{ wins}|PrivS' = PrivS]\Pr[PrivS' = PrivS] + \\
& \Pr[A \text{ wins}|s = s_1 \neq s_2 \neq \ldots \neq s_l \wedge PrivS' \neq PrivS]\times \\
& \Pr[s = s_1 \neq s_2 \neq \ldots \neq s_l \wedge PrivS' \neq PrivS] \\
\leq {} & 1\left(\tfrac{1}{|SSPACE|}\right) + 1\left(\tfrac{q_F}{|PSPACE|}\right) + \tfrac{1}{2}\left(1 - \tfrac{1}{|SSPACE|} - \tfrac{q_F}{|PSPACE|}\right) \\
\leq {} & \tfrac{1}{2} + \tfrac{1}{2|SSPACE|} + \tfrac{q_F}{2|PSPACE|}, \\
& \text{where } \epsilon = \tfrac{1}{2|SSPACE|} + \tfrac{q_F}{2|PSPACE|} \text{ is negligible.}
\end{aligned}$$

A has negligible 'advantage' in making random guessing the challenge output. $KDF4$ is secure in KPM, KPS, CCM and CCS.

Secondly, we show that KDF4 is not CPM-secure ($q = 0, l > 1$). When $q = 0$, A is not allowed to play at the learning stage and at the adaptive stage. In the challenge stage, the challenge key is computed as $K' = KDF4(PrivS, s, PrivS)_n$, where s is chosen by A. F is modeled as a RO, A can only distinguish if K' is the key generated by $KDF4$ or a random string of the same length, only if, A chooses $s = s_1\|s_2\|\ldots\|s_l$, $s_1 = s_2 = \ldots = s_l$, such as $K' = F(PrivS\|s_1\|PrivS)$ $\|\ldots\|F(PrivS\|s_l\|PrivS)$ and $F(PrivS\|s_1\|PrivS) = \ldots = F(PrivS\|s_l\|PrivS)$. A can distinguish K' is the cryptographic key when A observes there is a l repetition component parts. Since s is chosen by A, hence the probability is $\Pr[s_1 = s_2 = \ldots = s_l] = 1$. Therefore, A can distinguish between a derived cryptographic key and a truly random string based on the observation of the l repetition component parts. Thus, $KDF4$ is not CPM-secure. □

KDF5 Here we analyse the security of *KDF5*, which corresponds to the KDF proposal presented by Krawczyk in [6]. *KDF5* is defined as,

$$KDF5(PrivS,\ s,\ ctx)_n = G(F(PrivS,\ s),\ ctx)$$

In the following analysis, both G and H are modelled as random oracles, and we show that $KDF5$ is CPM-secure.

Remark: We must note that Krawczyk [6] proved that $KDF5$ is CCS-secure in the standard model. Specifically, Krawczyk proves that if F is a good (randomised) computational extractor and G is a pseudorandom function, then the composition shown above is CCS-secure. He then goes on to show that the standard MAC algorithm HMAC satisfies both requirements under standard assumptions in the underlying hash function. Extending Krawczyk's result to CPM security in the standard model would necessitate the modification of the given definition of computational extractor, to relax the requirement on the salt being chosen uniformly at random. This notion of extractor would be trivially satisfied

by deterministic extractors, such as those mentioned by Krawczyk himself [6]. We leave the formalisation of these changes as future work and now focus in showing that CPM-security is achievable in the ROM.

Theorem 5 *KDF5 is secure with respect to all five security models.*

Proof: Firstly, we show that *KDF5* [6] is CPM-secure. During the learning stage, C chooses $PrivS \in_R PSPACE$. C uses F to compute $K_i = KDF5(PrivS, s_i, ctx_i)$, where s_i and ctx_i are chosen by A from $SSPACE$ and $CSPACE$ respectively. A is provided K_i. In the challenge stage, challenge key is computed as $K' = KDF5(PrivS, s, ctx)$, where s and ctx are chosen by A. C sends K' to A, A continues learn the cryptographic keys up to $q - q'$ queries.

Since F is modeled as a RO, A can only distinguish if K' is the key generated by *KDF5* or a random string of the same length, only if:

a) $(s, ctx) = (s_i, ctx_i)$ for some $i = 1 \dots q_k$, it means, $G(F(PrivS,\ s),\ ctx) = G(F(PrivS,\ s_i),\ ctx_i)$. A will distinguish K' is one of the key at the learning stage. However, based on Definition 5, the chosen pair $(s,\ ctx)$ is restricted not the similar pair $(s_i,\ ctx_i)\ \forall\ i$ in the learning stage. Hence, the probability is $\Pr[(s, ctx) = (s_i, ctx_i)] = 0$.
b) Query $PrivS$ such as $PrivS' = PrivS$ and find $F(PrivS'\|s) = F(PrivS\|s)$ or $F(PrivS'\|s\|ctx) = F(PrivS\|s\|ctx)$. This implies that A guessing $PrivS$ with with probability $Pr[PrivS' = PrivS] \leq \frac{q_F}{|PSPACE|}$.

The probability that A winning this indistinguishability game is:

$$\begin{aligned}\Pr[A \text{ wins}] = &\Pr[A \text{ wins}|PrivS' = PrivS]\Pr[PrivS' = PrivS] + \\ &\Pr[A \text{ wins}|PrivS' \neq PrivS]\Pr[PrivS' \neq PrivS] \\ &\leq 1\left(\frac{q_F}{|PSPACE|}\right) + \frac{1}{2}\left(1 - \frac{q_F}{|PSPACE|}\right) \\ &\leq \frac{1}{2} + \frac{q_F}{2|PSPACE|}, \text{ where } \epsilon = \frac{q_F}{2|PSPACE|} \text{ is negligible.}\end{aligned}$$

A is only has negligible 'advantage' to distinguish the challenge output. Therefore, *KDF5* is CPM-secure. Hence *KDF5* is secure in CCM, KPM, CCS and KPS by Lemma 1, Lemma 2, Lemma 3 and Lemma 4 respectively. □

6 Conclusion

We propose four new security models known as KPM, KPS, CCM and CPM. Together with the CCS security model (proposed by Krawczyk), we believe these security models form a comprehensive security framework for KDFs. This allows for consideration of the security of a KDF against adversaries of varying capabilities. We establish the relations between these five security models. These relations are established using existing and modified KDF proposals.

As future work, we plan to use our proposed security framework to analyse the security level of other existing KDF proposals and, where possible, extend our analyses to the standard model (i.e without random oracles).

References

1. C. Adams, G. Kramer, S. Mister, and R. Zuccherato. On the security of key derivation functions. *Information Security*, 3225:134–145, 2004.
2. B. Barak, R. Shaltiel, and E. Tromer. True random number generators secure in a changing environment. *Cryptographic Hardware and Embedded Systems-CHES 2003*, pages 166–180, 2003.
3. Q. Dang. Recommendation for Existing Application-Specific Key Derivation Functions. *NIST Special Publication*, 800:135, 2010.
4. B. Kaliski. PKCS# 5: Password-based cryptography specification version 2.0. Technical report, RFC 2898, September 2000, 2000.
5. H. Krawczyk. On Extract-then-Expand Key Derivation Functions and an HMAC-based KDF, 2008. Online available at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.8254&rep=rep1&type=pdf.
6. H. Krawczyk. Cryptographic Extraction and Key Derivation: The HKDF Scheme. *Advances in Cryptology–CRYPTO 2010*, pages 631–648, 2010.
7. H. Krawczyk and P. Eronen. HMAC-based Extract-and-Expand Key Derivation Function (HKDF). Technical report, RFC 5869, May 2010, 2010.
8. D. McGrew and B. Weis. Key Derivation Functions and Their Uses, 2010. Online available at url http://www.ietf.org/id/draft-irtf-cfrg-kdf-uses-00.txt.
9. V. Shoup. ISO 18033-2: An emerging standard for public-key encryption. *Final Committee Draft (December 2004)*, 2004.
10. F.F. Yao and Y.L. Yin. Design and analysis of password-based key derivation functions. *Topics in Cryptology–CT-RSA 2005*, pages 245–261, 2005.